Integrating Reinforcement AI Into the Design of Educational Games

Ashish Amresh
Northern Arizona University, Flagstaff, USA
ashish.amresh@nau.edu

Abstract: The emergence of reinforcement-based AI for text generation (Chat-GPT) and image creation (Dall-E) has opened a wide range of possibilities for changing the game design and development process. While game development researchers have mostly focused on integrating these technologies to improve production workflow and demonstrate their use in the creation of content for entertainment purposes (intelligent NPCS), there is very little knowledge on how to integrate this technology into the design of educational games. In this paper, we present the results of integrating reinforcement AI (text and image generation) into educational gaming experiences by graduate students enrolled in a game-based learning course. The students were given a core set of requirements that enable the integration into their project with some flexibility on the desired educational outcome. The produced experiences were then evaluated by a small sample of experts (gaming and learning sciences) and their observations were compiled. Specifically, we describe the wide range of experiences developed by the students and the results of a qualitative study with a small group of experts that evaluated these experiences. Our results indicate that reinforcement AI-based integrations into educational game design and development helps enrich the user experience and has the potential to improve learning outcomes.

Keywords: Educational game design, Reinforcement AI, Text completion, Image generation

1. Introduction

Educational games are a large and profitable industry, estimated at about twenty million dollars in 2007 (Susi, Johannesson & Backlund, 2007). Due to the large amount of money at stake, educational game developers are looking to minimize costs and increase profit by making more games at a lower cost. A significant cost in the creation of games is associated with the content. Content can take many forms such as rich narrative text, dialogues and scripts, imagery, environments, props, objects, and animation to highlight a few. Large language models (LLMs) have shown potential in generating video game content, including characters, quests, dialogue, and even entire game narratives. These models use deep learning techniques to analyze and learn from vast amounts of text data and generate new content that is similar in style and tone to the original data.

Reinforcement learning (RL) is a type of artificial intelligence (AI) that involves training an agent to make decisions in an environment by rewarding or punishing it based on its actions. Large language models (LLMs) can be used in reinforcement learning to provide the agent with a natural language interface to the environment it is interacting with. One application of reinforcement learning with LLMs is in natural language understanding and generation. The LLM can be trained to understand natural language input from the user and generate natural language responses, allowing the agent to communicate with humans in a natural and intuitive way. Another application is in game playing, where RL algorithms can learn to play games such as chess, Go, or video games, by interacting with the game environment and receiving rewards or penalties based on their actions. LLMs can be used to provide the agent with additional context or information about the game environment, such as character descriptions or game lore, which can help the agent make more informed decisions.

LLMs can assist in-game content generation by generating NPC dialogue. NPCs, or non-playable characters, are characters in a video game that are controlled by the game's AI rather than the player. These characters often have scripted dialogue that is pre-written by game developers. By training a large language model on existing NPC dialogue, it can learn to generate new dialogue that is consistent with the game's style and tone. Another application is generating quests or storylines for the game. By training a language model on a large corpus of fantasy or sci-fi literature, for example, it can learn to generate new stories and quests that fit within the game's world and narrative. This approach can help game developers create more content in less time and generate new and creative ideas for quests and storylines.

Finally, LLMs can be used to generate game characters, including their backstories, personalities, and motivations. By training on a large corpus of character descriptions from literature or other media, the model can generate new and unique character concepts that can be used in a game. Overall, large language models have the potential to be a powerful tool for video game content generation, providing game developers with new and creative content ideas and reducing the time and cost associated with creating game content from scratch.

These needs leave us with three research questions:

RQ1. To what extent can RL assist in the educational game design process?

RQ2. What is the role of the content creator or producer when using RL during content creation?

This paper aims to provide answers to these two research questions. First, this paper discusses current research in both RL and LLMs and their application in game development. This is explored further by having students integrate OpenAI and the Unity game engine to build representative educational game prototypes. These prototypes are then evaluated by experts and a survey on the efficacy of using auto-generated game content is administered to get some insights into the research questions.

2. Current State of the Art

Procedural content generation (PCG) has been an emerging area in game development and particularly the appeal lies in its use of algorithms to create game content such as NPC dialogue, levels for puzzles, and sometimes entire worlds without human intervention (Liu et al., 2021). PCG provides dynamism and unpredictability to the game experience as players are not completely sure what the nature of the content and level progression can be. PCG has been applied to the game design process in several ways and the most common approaches have been 1) Randomization – where the content is generated randomly with a loose description of the parameters; 2) Rule-driven – where the content is generated by a defined set of rules and provides more structured content; 3) Evolutionary – where the content is generated based on what the player does in the game and provides greater dynamism and richness to the experience. Some of the common examples of PCG use in popular games include Minecraft, Rogue Legacy, Spelunky, and No Man's Sky where PCG has been typically used to generate the worlds and levels in the game.

LLMs have the potential to expand the use of PCG for game content that is based on narration and dialogue for use within the educational game context. Training an LLM on existing educational content and using the trained model to produce rich and real-time text generation for educational games has been achieved in the context of building dynamic quizzes and tests. The trained models also have demonstrated the ability to adapt to the learner by adjusting and changing the nature of the content to suit their needs. Todd et al. (2023) recently show the use of GPT-3 for generating a level for a puzzle game *Sokoban* and conduct a study where participants cannot distinguish between human and AI-generated content. Ashby et al. (2023) introduce the idea of generating quest-based dialogue within an RPG game using an LLM approach. They evaluate and demonstrate that the quests and the resulting dialogue match the performance of human-crafted quests in terms of fluency, coherence, and creativity. Story and narrative generation are also demonstrated via the Wordcraft tool which assists users in writing stories and provides better assistance in producing creative and novel narratives compared with not using the tool (Yuan et al., 2022). To further illustrate the growing interest in the area of applying LLMs for game content generation, the 2023 IEEE Conference on Games ran the first ChatGPT4CG competition where teams had to develop AI that creates character-like levels for the game *Science Birds* (Taveekitworachai et al., 2023). The competition was evaluated on stability and similarity to human-crafted characters.

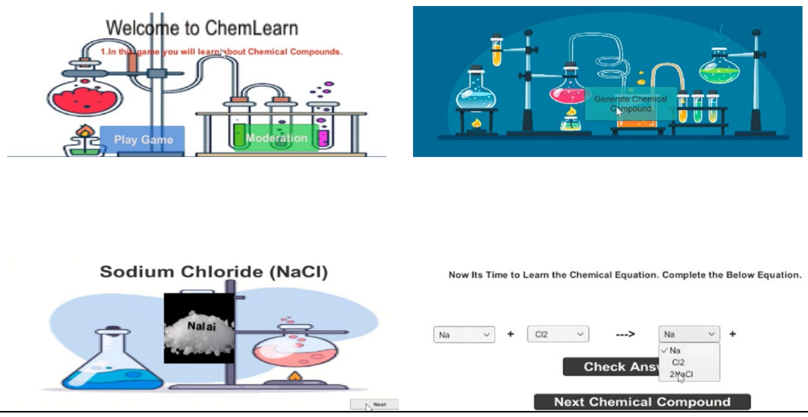
The majority of the experiments that study the integration of LLMs with game content creation use OpenAI with the Unity game engine. OpenAI provides access to a variety of AI models, including GPT-3, which can be used to generate text, translate languages, write different kinds of creative content, and answer questions in an informative way. When used together, OpenAI and Unity can be used to create games that are engaging, interactive, and intelligent. For example, OpenAI's GPT-3 can be used to generate text-based content for games, such as dialogue, descriptions, and even entire stories. Unity can then be used to create the visual elements of the game, such as the characters, environments, and gameplay. In *Ava: Children of the Machine* GPT-3 is used to generate text-based content, such as dialogue and descriptions. In *The Bard's Tale IV: Barrows Deep* GPT-3 is used to create AI characters that are more intelligent and realistic and in *Project Wingman* GPT-3 is used to create chatbots that players can interact with. While relatively a new field, GPT-3 and Unity have also been combined to develop educational game content in language learning (Zografos and Moussiades, 2023). With demonstrated success in creating human-like content for game development, there is indeed great enthusiasm in the development community for continuing to improve the integration of tools such as OpenAI and Unity which will lead to newer research and development efforts in studying the role of LLMs for developing rich, engaging, and meaningful content for educational games. In the next section, we present our experiment that starts the process of thinking about the integration of OpenAI and Unity and demonstrates their capabilities by developing simple educational game prototypes.

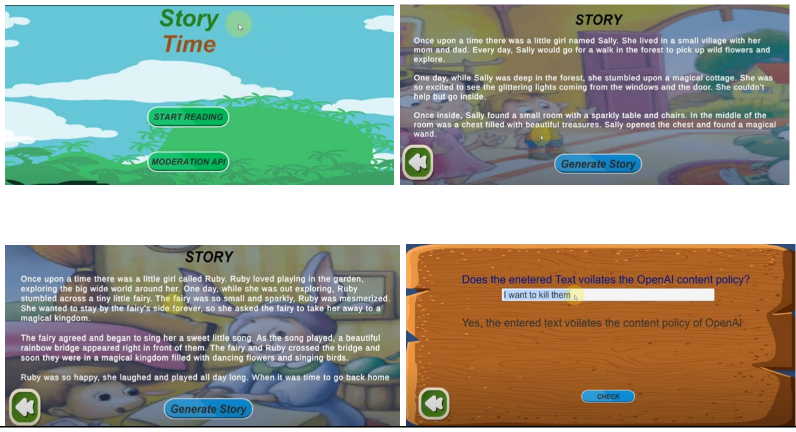
3. Experimental Design

For this study, 10 computer science graduate students enrolled in a game-based learning course were provided a project to complete as part of their course activities. They were provided an OpenAI API Unity asset and asked to create a new Unity 3D component based on this asset. Once the component was created, they could add this to any game object in their game and use GPT-3 for text-based information generation and DALL-E for an image-based generation. The steps that were necessary to complete the project include: 1) Setup – where they would install the provided asset, create an OpenAI account with a provided organization ID, create an API key, and verify the credentials via the Unity editor. Once setup is complete Unity and OpenAI can communicate and the requests from the game via the OpenAI API would get validated; 2) Review existing components – the downloaded asset comes with two example components *OpenAIImageReplace* and *OpenAiTextReplace* that can be added to any Unity game object in the scene and tested to check if text and image generation are working; 3) Review existing code – students are asked to review the provided code samples for the two components and also review the OpenAI API documentation and learn how the integration with Unity works; and lastly 4) Create something new – students were asked to create a new component that has an educational objective such as using the text generation to create narrative pieces or the image generation to create new game textures or the combination of the two. Some initial ideas that were proposed include the creation of a children's storybook, building a dialogue system that helps practice math problems, and creating an interactive and dynamic quiz on a particular chosen topic. Students were asked to challenge the limits and be creative.

Students had three weeks to complete the above steps and demonstrate their work. Table 1 lists some sample projects that were created by the students. The paper presents some of these projects as there was considerable similarity in the chosen ideas for the project among the students.

Table 1: Sample projects produced during the course

Objectives and goals	Screenshots
Teach chemical compounds – GPT-3 generates the compounds and the element composition on the fly; DALL-E generates the compound image. Students are given AI-generated compounds, with representative images and they must create the correct bonding equation to progress in the game.	
A *Jeopardy-style* game where the player selects a topic, and the questions and answers are generated by OpenAI. Topics include history, science, and social studies. Additionally, has a Chatbot where you can ask questions or get help with math.	

<table>
<tr><th>Objectives and goals</th><th>Screenshots</th></tr>
<tr><td>A storybook game for children where the content is generated by OpenAI including the text and images. Includes a moderation API so that explicit or poor content can be filtered before displaying it to children. The game generates new stories and images while moderating the content each time the player clicks the button.</td><td>
</td></tr>
</table>

The course had a final team project that tasked the students to learn the concepts and create an educational game prototype. Among the three project teams, one team chose to integrate OpenAI with their final project game titled *Traffic Learner*. The game was aimed at those learning to drive a vehicle and familiarizing themselves with traffic rules and regulations. The game had three major components: 1) Information – learn and familiarize with traffic rules and regulations, the content and imagery were generated via OpenAI; 2) Quiz – test the knowledge of the rules and regulations, the questions and answers were generated by OpenAI; and 3) Play – practice driving in a real-world setting by following the rules and regulations. Figure 1 shows the three modes developed for the game.

Figure 1: Information (left), Quiz (middle), and Play (right) areas of the Traffic Learner game

4. Findings

Two faculty members evaluated the effectiveness of integrating OpenAI and Unity in the student projects. One faculty member was an expert in human-computer interaction (HCI) and the second faculty member was an expert in software engineering (SE). This helped the evaluation be conducted from multiple viewpoints and lenses. The feedback was gathered via a data questionnaire that the members filled out during student project presentations. The HCI faculty member raised questions about the trust in the content and the possibility of misguiding the player. The SE faculty member raised questions about the content's replicability and consistency. They were also unsure about issues relating to copyright and ownership, especially with the auto-generated imagery. Overall, both faculty members were supportive of exploring ways to integrate OpenAI with Unity and building educational game experiences. This provides a positive direction for our first research question on the role of OpenAI in assisting with educational game design.

OpenAI and Unity integration also had many challenges that would need to be overcome in the future. Time inconsistency was the most common challenge, this relates to the time it takes OpenAI to generate the content. The factors that influence this include the load on OpenAi's servers, the nature of the content desired, and the bandwidth availability during play. Potentially, players may have to wait for seconds at large to have OpenAI content be produced and displayed while playing the game. Some suggestions include pre-querying methods for generating and storing the content and making it available via a just-in-time approach. This requires prior knowledge of what the game's content needs are and may not always work when the environment is rapidly evolving. For example, if a player needs to talk to an NPC and the content produced for the dialogue depends on what the player asks then a just-in-time approach may not work. However, if the question set is limited and well-known prior to the query then the possible answers could be populated in advance. There is limited understanding of our second research question on the role of humans in developing a game that uses OpenAI-generated content. While we cannot completely rely on OpenAI content in all situations, there are positive use cases where a collaboration between a human and AI is necessary for generating desirable educational game content. Further research in this area is necessary to explore the right circumstances and situations for tight collaboration. It is also foreseeable that advancements in the text and image generation methods within OpenAI and other LLM research projects could provide the framework for generating and integrating content without any human intervention.

5. Future Work

There is real potential for the integration of AI-generated content in educational games. When the objective is to build proficiency in a particular topic and speed of development becomes necessary, then such an integration can be valuable for educational game designers and developers. In our prior work, we demonstrated an approach to being content-agnostic when rapidly designing educational games via the content-agnostic game engineering (CAGE) framework (Baron et al., 2016). AI-generated content can play an important role in how CAGE (Verma et al., 2021) can be further developed to add greater agility and speed in development. Stealth assessment techniques (Verma et al., 2019) can further be enhanced and response mechanisms that manage the dynamism in the content can be efficiently handled. Figure 2 shows the future modification to the CAGE framework and would form the basis for future game studies in closing the loop between AI-generated content, agnostic content, and stealth assessment. This would allow for the game to decide in real time whether to pull in static inactive content and make that active or reach out to OpenAI and ask for AI-generated content. The nature of the AI-generated content could additionally be tied with the mechanics component and based on the game mechanic (narration, puzzle, level, etc.) appropriate queries could be handled by OpenAI. The stealth assessment aspects of the framework (in yellow) would be able to provide an additional layer of information back to the system on whether the content generated by the AI would need to be changed or updated leading to a cyclical evolutionary approach to the game.

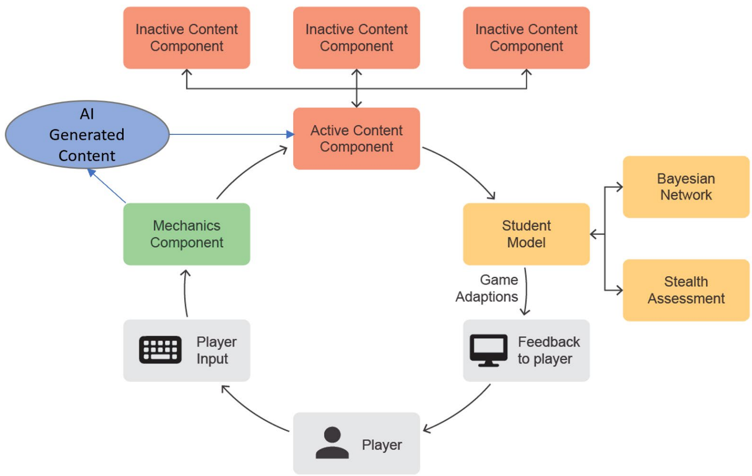


Figure 2: The modification (in Blue) to the CAGE framework

6. Conclusion

In this paper, we introduced the integration of reinforcement learning AI techniques that apply large language models within the context of educational game design and development. We compiled and presented the products developed by students and student teams when tasked with integrating OpenAI and the Unity game engine within the context of a graduate-level game-based learning course. Our findings provide early insights into the ability of reinforcement learning techniques in assisting the process of educational game development and the role that human content creators would play when employing such techniques. There are several limitations with advancing the use of AI-generated content within educational games. Quality control of the generated content can be limiting, for example the content that is rule based can such as chemical compounds can be reliably presented with the possibility of adding additional constraints such as safety of the resulting mixture. Another possible concern is being able to closely match the generated text and images so that the resulting experience does not appear to be ambiguous to the player. As research in LLM matures and connectivity across openAI tools increases, the potential for risks associated with trust, validity and ambiguity of the generated content have the potential to reduce.

References

Ashby, T., Webb, B.K., Knapp, G., Searle, J. and Fulda, N., 2023, April. Personalized Quest and Dialogue Generation in Role-Playing Games: A Knowledge Graph-and Language Model-based Approach. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1-20).

Baron, T., Heath, C. and Amresh, A., 2016, October. Towards a context agnostic platform for design and assessment of educational games. In *European conference on games based learning* (p. 34). Academic Conferences International Limited.

Liu, J., Snodgrass, S., Khalifa, A., Risi, S., Yannakakis, G.N. and Togelius, J., 2021. Deep learning for procedural content generation. *Neural Computing and Applications*, *33*(1), pp.19-37.

Susi, T., Johannesson, M. and Backlund, P., 2007. Serious games: An overview.

Todd, G., Earle, S., Nasir, M.U., Green, M.C. and Togelius, J., 2023, April. Level Generation Through Large Language Models. In *Proceedings of the 18th International Conference on the Foundations of Digital Games* (pp. 1-8).

Taveekitworachai, P., Abdullah, F., Dewantoro, M.F., Thawonmas, R., Togelius, J. and Renz, J., 2023. ChatGPT4PCG Competition: Character-like Level Generation for Science Birds. *arXiv preprint arXiv:2303.15662*.

Verma, V., Baron, T., Bansal, A. and Amresh, A., 2019. Emerging practices in game-based assessment. *Game-based assessment revisited*, pp.327-346.

Verma, V., Craig, S.D., Amresh, A. and Bansal, A., 2022. Content Agnostic Game Engineering: Impact of Stealth Assessment and Content Order on Player Engagement. In *Proceedings of the Future Technologies Conference (FTC) 2021, Volume 3* (pp. 455-470). Springer International Publishing.

Verma, V., Amresh, A., Craig, S.D. and Bansal, A., 2021. Validity of a Content Agnostic Game Based Stealth Assessment. In *Games and Learning Alliance: 10th International Conference, GALA 2021, La Spezia, Italy, December 1–2, 2021, Proceedings 10* (pp. 121-130). Springer International Publishing.

Yuan, A., Coenen, A., Reif, E. and Ippolito, D., 2022, March. Wordcraft: story writing with large language models. In *27th International Conference on Intelligent User Interfaces* (pp. 841-852).

Zografos, G. and Moussiades, L., 2023, May. A GPT-Based Vocabulary Tutor. In *Augmented Intelligence and Intelligent Tutoring Systems: 19th International Conference, ITS 2023, Corfu, Greece, June 2–5, 2023, Proceedings* (pp. 270-280). Cham: Springer Nature Switzerland.